

BAB III METODE PENELITIAN

3.1 Alat Dan Bahan

Perangkat-perangkat yang digunakan pada penelitian ini dibagi menjadi dua, yaitu perangkat keras (*hardware*) berupa tiga router Cisco yang dihubungkan menggunakan tipe kabel UTP Cat5e *Crossover* dan kabel UTP Cat5e *Straight* untuk dihubungkan ke-empat PC dengan sistem operasi *windows* yang dihubungkan dengan *router* dan satu laptop menggunakan kabel *console* untuk *remote control* router Cisco 2800 *series* dan 2 router Cisco 1800 *series*. Berikut tabel 3.1 adalah spesifikasi perangkat keras yang digunakan.

Tabel 3.1 Spesifikasi perangkat keras

Laptop Server 1	Processor	Intel Core i5-4200M @ 2.50 GHz (4 CPUs)
	RAM	4096 MB DDR3L
	Hardisk	1TB 5200 RPM
	NIC	Realtek PCI-E <i>Fast Ethernet Controller</i>
Laptop Client 2	Processor	AMD Celereon @ 2.50 GHz (4 CPUs)
	RAM	2048 MB DDR3
	Hardisk	500 GB 5200 RPM
	NIC	Realtek PCI-E <i>Fast Ethernet Controller</i>
PC Server 3 dan Client 4	Processor	Intel Core i3-4100U @ 2.40 GHz (4 CPUs)
	RAM	8192 MB DDR3
	Hardisk	500 GB 5200 RPM
	NIC	Atheros AR8152 PCI-E <i>Gigabite Ethernet Controller</i>
Router 2800 Series	LAN Interface	10/100-Mbps <i>Fast Ethernet</i>
Router 1800 Series	LAN Interface	10/100-Mbps <i>Fast Ethernet</i>

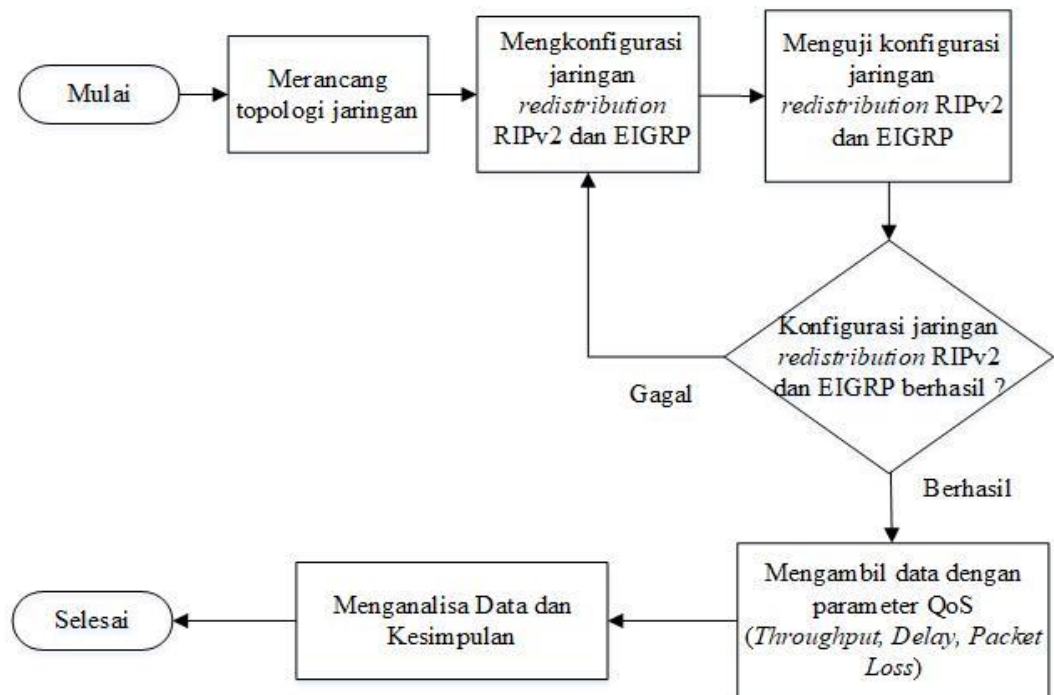
Lalu perangkat lunak (*software*) yang dibutuhkan dalam penelitian ini seperti berikut :

1. *Software Putty software* digunakan untuk *remote control* pada *router* Cisco.
2. *File Sharing Windows* sebagai media *File Transfer* untuk berbagi *file* antar *server* ke *client*.

3. VLC media *player* merupakan *software* pemutar media seperti audio atau video.
4. Wireshark digunakan sebagai *network analyzer*, yaitu untuk memonitoring paket-paket pada sebuah jaringan.

3.2 Alur Penelitian

Dalam melakukan penelitian ini, terdapat beberapa tahapan berikut pada gambar 3.1 merupakan *flowchart* dari proses penelitian ini.



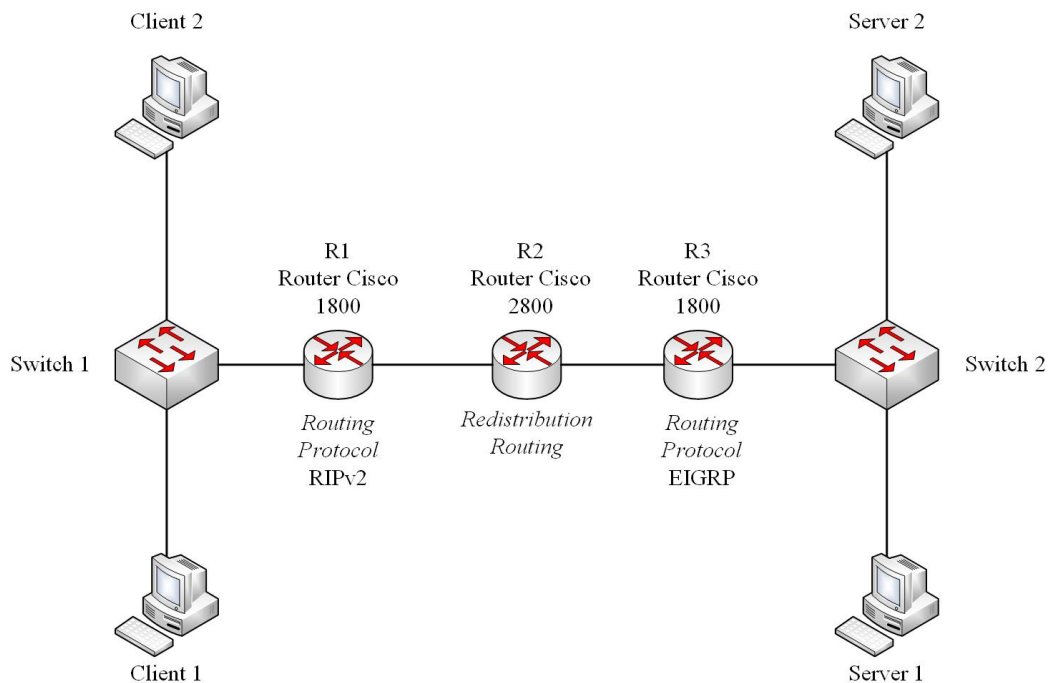
Gambar 3.2 *Flow Chart* Penelitian

Pada gambar 3.2 merupakan alur kerja dari pengerjaan penelitian secara keseluruhan. Pada alur pengerjaan ini menggambarkan langkah-langkah untuk melakukan analisa proses redistribus *routing protocol* RIPv2 dan EIGRP. Pertama adalah merancang topologi jaringan. Disini akan digunakan tiga router Cisco, dua PC dan, dua laptop. Setelah perancangan topologi sukses, langkah selanjutnya adalah konfigurasi topologi yang menggunakan IPv4 pada setiap router dengan konfigurasi *routing protocol* RIPv2 dan EIGRP menggunakan metode redistribusi. Setelah sukses mengkonfigurasi topologi, selanjutnya adalah menguji konfigurasi jaringan redistribusi pada *routing protocol* RIPv2 dan EIGRP dengan melakukan tes

ping. Jika dalam proses menguji jaringan redistribusi gagal maka akan kembali pada tahap mengkonfigurasi jaringan redistribusi, jika berhasil maka dilakukan pengambilan data pada tahap ini merupakan proses dimana topologi jaringan pada sisi *server* menuju *client* menjalankan proses pengiriman data menggunakan *file sharing windows* dan *video streaming* yang dilanjutkan dengan pengambilan data QoS yaitu parameter-parameter yang akan di ujikan seperti *throughput*, *delay*, dan *packet loss*. Setelah itu semua telah selesai maka akan masuk pada tahap analisa dan kesimpulan maka selesai sudah penelitian ini.

3.3 Merancang Topologi Jaringan

Dalam merancang topologi jaringan, menggunakan topologi seperti pada gambar 3.2 berikut.



Gambar 3.2 Topologi Jaringan Redistribusi RIPv2 dan EIGRP

Pada gambar 3.2 menjelaskan tentang topologi jaringan dari penelitian yang akan dilakukan. Dalam topologi jaringan redistribusi RIPv2 dan EIGRP menggunakan dua laptop dan dua PC sebagai *client*, dan sebagai *server*, satu laptop sebagai *remote control* dan tiga *router* Cisco dimana masing-masing PC dan laptop menggunakan sistem operasi *windows*. IP

yang digunakan yaitu IPv4 sebagai pengalamatan *network* pada *routing protocol* RIPv2 dan EIGRP. Masing-masing router R1, R2, dan R3 dihubungkan. Pada R1 menggunakan router Cisco 1800 *series* dengan konfigurasi *routing protocol* RIPv2, R3 menggunakan router Cisco 2800 *series* dengan konfigurasi *routing protocol* EIGRP dan R2 menggunakan router Cisco 2800 *series* dengan konfigurasi *redistribution routing* antara *routing protocol* RIPv2 dan EIGRP. Dari gambar 3.2 topologi jaringan tersebut diberikan IP yang telah disiapkan seperti pada tabel 3.2 di bawah ini :

Tabel 3.2 Tabel Pengalamatan IP

Perangkat	IP	Alamat
<i>Laptop Client 1</i>	IPv4	10.0.0.11/24
<i>Laptop Server 1</i>	IPv4	20.0.0.12/24
<i>PC Client 2</i>	IPv4	20.0.0.14/24
<i>PC Server 2</i>	IPv4	10.0.0.13/24
<i>Router Cisco 1800</i>	IPv4	10.0.0.2/24
	IPv4	192.168.0.1/24
<i>Router Cisco 2800</i>	IPv4	192.168.0.2/24
	IPv4	192.168.1.1/24
<i>Router Cisco 1800</i>	IPv4	192.168.1.2/24
	IPv4	20.0.0.1/24

Pada tabel 3.2 menjelaskan pengalamatan IP yang digunakan pada gambar 3.2 untuk melakukan penelitian dimana pengalamatan IP yang digunakan adalah IPv4 untuk perangkat PC , laptop dan router yang dikonfigurasi menggunakan *routing protocol* RIPv2 dan EIGRP di topologi jaringan. Dalam penelitian ini menggunakan layanan pengiriman *file* menggunakan *file sharing windows* dan *video streaming* dengan metode redistribusi antara *routing protocol* RIPv2 dan EIGRP.

Untuk pengiriman *file* menggunakan layanan *file sharing windows* menggunakan skenario *background traffic* dan *no background*. Pada skenario *no background traffic*, melakukan pengiriman *file* dari *server 1* menuju *client 1* sedangkan *server 2* dan *client 2* tidak melakukan aktifitas

apapun. Pada skenario *background traffic*, melakukan pengiriman *file* dari *server* 1 menuju *client* 1 hanya saja terdapat aktifitas pengiriman *file* dari *server* 2 menuju *client* 2 sebesar 20GB. *File* yang digunakan berupa .zip/.rar dengan ukuran yang berbeda yaitu 25 MB, 50 MB, 75 MB, 100 MB, 125 MB, dan 150MB.

Untuk layanan *video streaming* menggunakan skenario *background traffic* dan *no background traffic*. Pada skenario *no background traffic* melakukan *video streaming* dari *server* 1 menuju *client* 1 sedangkan *server* 2 dan *client* 2 tidak melakukan aktifitas *video streaming*. Pada skenario *background traffic* melakukan *video streaming* dari *server* 1 menuju *client* 1 hanya saja terdapat aktifitas *video streaming* dari *server* 2 menuju *client* 2. *Video* yang digunakan menggunakan format .mp4 dengan durasi *video* 60 detik, resolusi *video* 1280×720 variasi *bit rate* berbeda yaitu 2000 kbps, 4000 kbps, 6000 kbps, 8000 kbps, 10000 kbps dan 12000 kbps.

3.4 Mengkonfigurasi Jaringan *Redistribution*

Router yang digunakan pada penelitian ini adalah *router cisco* seri seri 2600 satu unit, dan seri 1800 dua unit. Hal yang dibutuhkan dalam melakukan instalasi *router* adalah driver *router* dan *software Putty* serta kabel *console*. Driver *router* dibutuhkan untuk mendapatkan hak akses untuk mengkonfigurasi *router* melalui PC/laptop, sedangkan *software putty* digunakan untuk penghubung antara PC/laptop ke *router* yang akan dikonfigurasi melalui media transmisi kabel *console*.

Setelah *driver* terinstall dengan baik, maka selanjutnya adalah konfigurasi *software putty* agar PC/laptop dapat terhubung dengan *router*. *Software putty* adalah *software* yang digunakan sebagai penghubung komunikasi antara *router* dan PC/ laptop. Sebelum mengkonfigurasi *software putty* terlebih dahulu siapkan kabel *console* sebagai media transmisinya. Tahapan yang dilakukan dalam mengkonfigurasi *software putty* adalah sebagai berikut.

1. Menyambungkan *router cisco* dengan PC/laptop

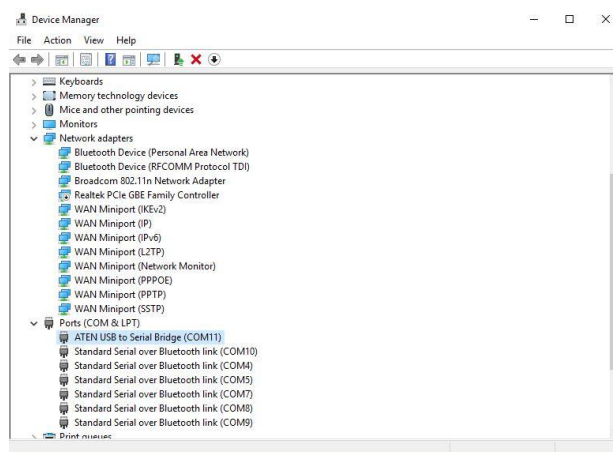
Untuk menyambungkan *router* dengan PC/laptop dibutuhkan kabel yang terdiri dari kabel console dan kabel *converter console to USB*.



Gambar 3.3 *router* yang tersambung dengan PC/laptop

2. Membuka device manager

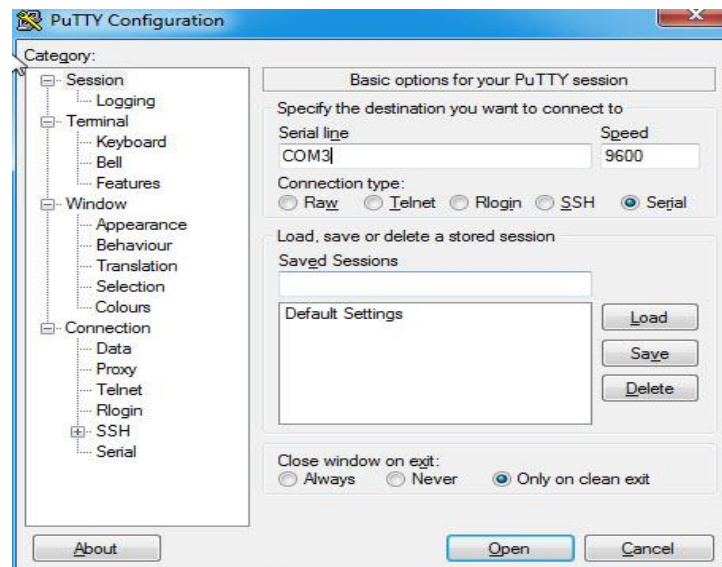
Setelah *router* dan PC/laptop tersambung, selanjutnya melihat nomor *port* yang digunakan. *Device manager* sendiri adalah fitur yang disediakan oleh *windows* untuk melihat dan mengatur *driver* yang telah masuk pada PC/laptop. Berdasarkan gambar 3.7 nomor *port* yang digunakan adalah COM11.



Gambar 3.4 tampilan *device manager*

3. Konfigurasi *software putty*

Setelah mengetahui nomor *port* yang akan digunakan, selanjutnya adalah membuka *software putty*. *Software* ini digunakan sebagai penghubung antara *router* dengan PC/laptop agar *router* dapat dikonfigurasi. Berikut adalah tampilan dari *software putty*.



Gambar 3.5 konfigurasi *software putty*

Pada menu *connection type*, pilih koneksi menggunakan *serial*, setelah itu isi nomor port COM11 yang digunakan pada kolom *serial line*. Perlu diperhatikan juga pada kolom *speed*. Untuk besaran *speed* yang digunakan, haruslah disesuaikan dengan *speed* yang bekerja pada *router* yang akan dikonfigurasi. Setelah semua parameter diisi dengan benar kemudian klik tombol open dan konfigurasi siap dijalankan.

Konfigurasi sistem menggunakan *software putty* v0.70 sebagai *remote control Router Cisco 2800 series dan 1800 series* untuk mengkonfigurasi *routing protocol RIPv2 dan EIGRP*. Implementasi berhasil ketika *server* dan *client* dapat melakukan tes ping. Untuk konfigurasi *routing protocol RIPv2 dan konfigurasi EIGRP menggunakan redistribution routing* pada *router Cisco 1800* pada Gambar 3.6 dan Gambar 3.7.

```
R1# configure terminal
R1(config)# router rip
R1(config-router)# version 2
R1(config-router)# network 10.0.0.0
R1(config-router)# network 192.168.0.0
R1(config-router)# no auto-summary
R1(config-router)# exit
```

Gambar 3.6 Konfigurasi routing protocol RIPv2

```
R3# configure terminal
R3(config)# router eigrp 100
R3(config-router)# network 10.0.0.0
R3(config-router)#network 192.168.0.0
R3(config-router)#no auto-summary
R3(config-router)# exit
```

Gambar 3.7 Konfigurasi routing protocol EIGRP

Konfigurasi kedua *routing* protokol diatas diterapkan pada semua *router*. Untuk alamat *network* bisa disesuaikan dengan rancangan jaringan yang akan digunakan. Pada gambar 3.6 dapat dilihat terdapat perintah `router rip`. Perintah `rip` disini ditujukan karena *router* akan dikonfigurasi dengan protokol `rip`. *Syntax network* berguna untuk menentukan alamat *network* yang terhubung langsung dengan *router* yang sedang dikonfigurasi. Sesuai dengan rancangan topologi jaringan, dalam penelitian ini kita hanya mengaktifkan protokol RIPv2 pada *port fast ethernet 0/0*. Untuk alamat *network* yang diketikan cukup bagian yang terhubung dengan *ethernet* tersebut saja.

Pada konfigurasi EIGRP seperti pada contoh gambar 3.7, *syntax router EIGRP* diikuti dengan angka 100, angka 100 disini merupakan nomer *autonomous system* (AS). Nilai AS yang diberikan bebas, namun untuk EIGRP nomer AS yang digunakan dalam jaringan harus sama agar dapat terhubung. Perintah `no auto-summary` digunakan untuk mengatasi sifat *classless* protokol EIGRP, yang mengakibatkan semua *network* berbeda yang berada dalam kelas yang sama, akan dibaca menjadi 1 *network* sesuai dengan kelas IP yang digunakan.

Routing protocol yang telah dikonfigurasi, belum dapat membuat setiap *router* yang ada dalam jaringan saling terhubung. Hal ini disebabkan karena *routing protocol* yang diterapkan dalam jaringan yang diteliti berbeda. Maka dilakukan konfigurasi redistribusi kedua *protocol routing*, seperti yang dapat dilihat pada gambar 3.8

```
router eigrp 100
 redistribute rip metric 100000 100 255 1 1500
 network 192.168.1.0
 no auto-summary
!
router rip
 version 2
 redistribute eigrp 100 metric 2
 network 192.168.0.0
```

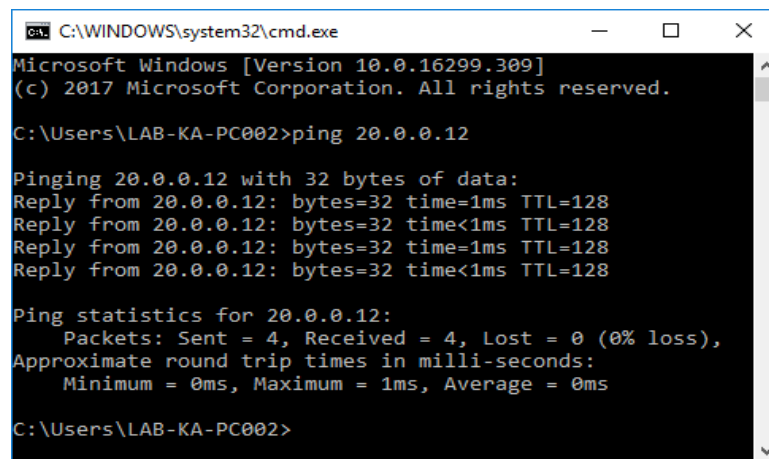
Gambar 3.8 Konfigurasi router 2 *redistribution routing* RIPv2 dan EIGRP

Pada gambar 3.8 dapat dilihat konfigurasi redistribusi protokol *routing* yang akan diterapkan ke setiap *router*. Sebelumnya masuk ke pengaturan *routing* EIGRP dengan mengetikkan *syntax* `router eigrp 100`. Begitu juga untuk masuk ke pengaturan *routing* OSPF dengan menggunakan *syntax* `router eigrp 100`. Untuk membuat informasi yang dibawa oleh protokol EIGRP dapat melewati protokol RIPv2, maka digunakan perintah `redistribute rip metric 100000 100 255 1 1500`. Nilai 100000 merupakan nilai *bandwidth metric* yang digunakan, dengan satuan Kbps. Nilai 100 menunjukkan nilai *delay metric* dalam satuan μ s. *Range* angka yang dapat digunakan untuk *bandwidth* dan *delay* ini yaitu antara 1-4294967295. Nilai 255 menunjukkan nilai *reliability metric* dengan *range* yang dapat digunakan antara 1-255. Semakin tinggi nilai yang kita masukan maka semakin *reliable*. Angka 1 pada gambar 3.11 diatas menunjukkan nilai *bandwidth effective*. *Range* yang dapat digunakan antara 1-255. *Metric* yang terakhir yaitu 1500 MTU (*Maximum Transmission Unit*) dengan *range* nilai yang dapat digunakan antara 1-65535. Untuk Protokol RIPv2 dapat melewati protokol EIGRP dengan menggunakan perintah `redistribute eigrp 100 metric 2`, angka 2 pada metric tersebut adalah *hop count*.

Pada gambar 3.8 menjelaskan konfigurasi jaringan router Cisco 2800. Dalam penelitian ini menggunakan dua skenario yaitu dengan *background traffic* dan *no background traffic*. Konfigurasi *metric* pada gambar 3.8 digunakan untuk menghubungkan *routing protocol* RIPv2 pada router 1 menggunakan router Cisco 1800 dan *routing protocol* EIGRP pada router Cisco 1800 pada router 3.

3.5 Menguji Konfigurasi Jaringan

Untuk menguji konfigurasi jaringan *redistribution* RIPv2 dan EIGRP pada masing-masing *client* dengan menggunakan IP yang telah ditentukan pada tabel 3.2, lakukan uji konektivitas dengan cara mengeping pada IP sumber ke IP tujuan maupun sebaliknya seperti Gambar 3.9 dan Gambar 3.10.



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.16299.309]
(c) 2017 Microsoft Corporation. All rights reserved.

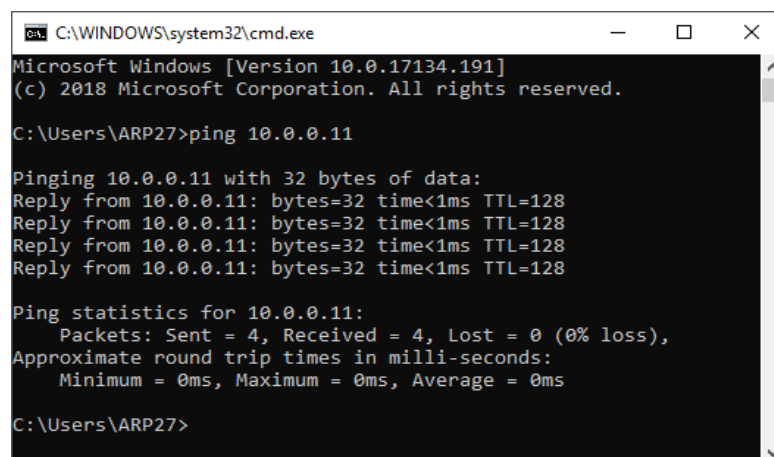
C:\Users\LAB-KA-PC002>ping 20.0.0.12

Pinging 20.0.0.12 with 32 bytes of data:
Reply from 20.0.0.12: bytes=32 time=1ms TTL=128
Reply from 20.0.0.12: bytes=32 time<1ms TTL=128
Reply from 20.0.0.12: bytes=32 time=1ms TTL=128
Reply from 20.0.0.12: bytes=32 time<1ms TTL=128

Ping statistics for 20.0.0.12:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\Users\LAB-KA-PC002>
```

Gambar 3.9 Uji konektivitas pada sisi *client*



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.17134.191]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\ARP27>ping 10.0.0.11

Pinging 10.0.0.11 with 32 bytes of data:
Reply from 10.0.0.11: bytes=32 time<1ms TTL=128
Reply from 10.0.0.11: bytes=32 time<1ms TTL=128
Reply from 10.0.0.11: bytes=32 time<1ms TTL=128
Reply from 10.0.0.11: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.11:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\ARP27>
```

Gambar 3.10 Uji konektivitas pada sisi *server*

Setelah mendapatkan informasi alamat IP milik *server* dan *client*, dilakukan pengujian konektivitas dengan melakukan ping di sisi *server* maupun di sisi *client*. Pada Gambar 3.4 terlihat bahwa *client* melakukan ping ke alamat IP *server* 20.0.0.12 dan pada Gambar 3.5 *server* melakukan ping ke alamat IP *client* 10.0.0.11.

3.6 Mengambil Data dengan Parameter QoS

Dalam penelitian ini proses mengambil data menggunakan pengiriman data yang berbeda, berupa *file* dan video. Untuk nilai parameter yang akan dianalisa yaitu *throughput*, *delay* dan *packet loss*. Proses pengambilan data menggunakan *software wireshark* untuk meng-*capture protocol* TCP untuk *file sharing*. Pengambilan data *file sharing* dilakukan dengan ukuran *file* seperti tabel 3.3 berikut.

Tabel 3.3 Rancangan skenario *File Sharing*

<i>File Size</i> (MB)	Parameter QoS
25	<i>Throughput</i> , <i>Delay</i> , <i>Packet loss</i>
50	
75	
100	
125	
150	

Pada tabel 3.3 merupakan rancangan skenario variasi ukuran *file*. Untuk skenario *no background traffic*, pengiriman *file* menggunakan *file sharing windows* dari *server* 1 menuju *client* 1 sedangkan *server* 2 menuju *client* 2 tidak ada aktifitas. Pada skenario *background traffic*, melakukan pengiriman *file* dari *server* 1 menuju *client* 1 hanya saja terdapat aktifitas pengiriman *file* dari *server* 2 menuju *client* 2 sebesar 20GB *File* data berupa .zip/.rar yang dikirimkan memiliki ukuran yang berbeda yaitu 25 MB, 50 MB, 75 MB, 100MB, 125MB, dan 150MB. Pada saat melakukan proses pengiriman *file*, *server* 1 dan *client* 1 sudah diaktifkan terlebih dahulu *software wireshark*. Dengan *software wireshark* nantinya akan tertangkap semua aktifitas *capture data*.

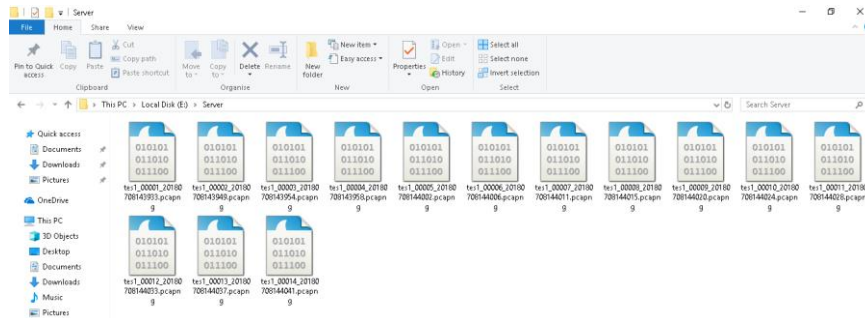
Untuk pengambilan data video menggunakan *software* pemutar video VLC dimana untuk pengaturannya jenis *transport protocol* yang dapat diatur secara manual, dalam hal ini *transport protocol* yang digunakan adalah RTP dengan meng-*capture protocol* UDP. Pengambilan data *video streaming* melakukan perubahan variasi *bit rate* seperti tabel 3.4.

Tabel 3.4 Rancangan skenario *video streaming*

Waktu	Resolusi Video	Bit rate (kbps)	Parameter QoS
60 detik	1280×720	2000	Throughput , Delay, Packet loss
		4000	
		6000	
		8000	
		10000	
		12000	

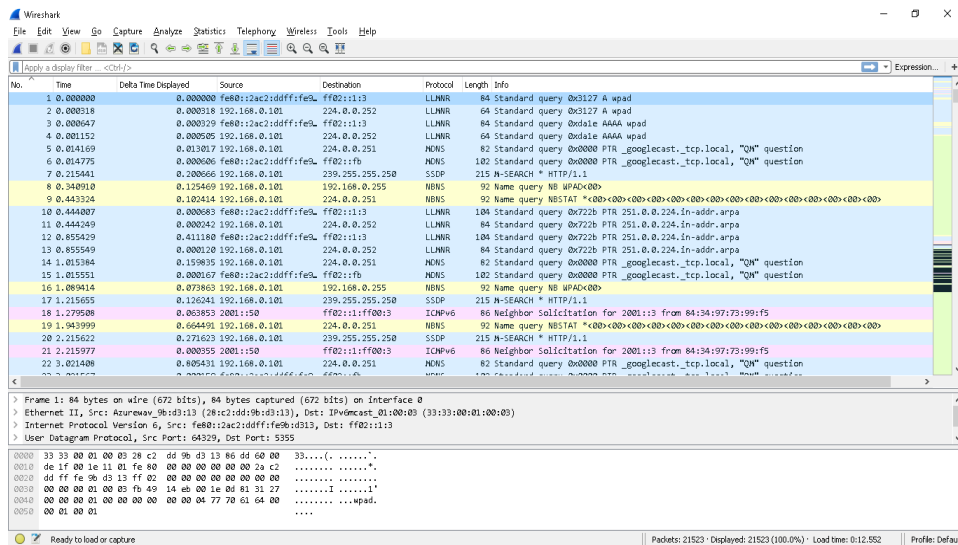
Pada tabel 3.4 layanan *video streaming* menggunakan tipe data berupa *video* dengan format .mp4 dengan durasi video 60 detik. Resolusi video yang digunakan adalah 1280×720, dengan *bitrate* berbeda-beda yaitu 2000 kbps, 4000 kbps, 6000 kbps dan 8000 kbps, 10000 kbps dan 12000 kbps. Untuk skenario *no background traffic* proses pengambilan data *video streaming* dari *server* 1 menuju *client* 1 dengan menggunakan *software wireshark* untuk pengambilan data. Sedangkan untuk skenario *background traffic* dilakukan oleh *server* 2 ke *client* 2 untuk pengiriman *file* sebesar 20GB dan melakukan *video streaming* untuk melihat perfomansi parameter QoS jika dalam suatu jaringan terdapat *traffic* atau jalur yang sedang digunakan oleh *server* 2 emenuju *client* 2. Dengan menggunakan *software wireshark* nantinya akan tertangkap semua aktivitas *capture* data. Selanjutnya adalah mengumpulkan hasil data dan membuat analisisnya.

Jika proses *capture* telah diberhentikan, maka hasil *capture* akan otomatis tersimpan dalam folder yang telah ditentukan seperti pada Gambar 3.11.



Gambar 3.11 Hasil Keluaran Dumpcap

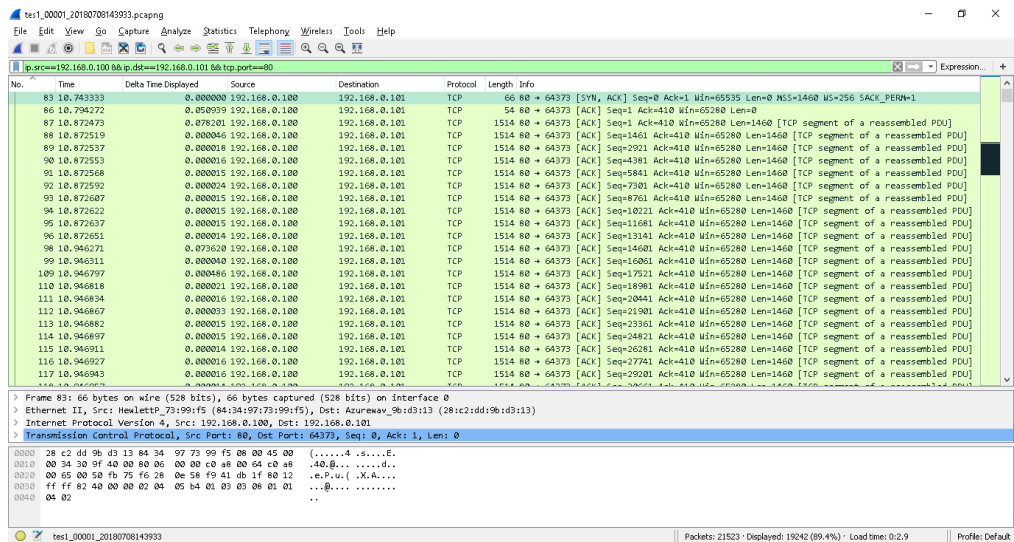
Hasil *capture* yang dihasilkan oleh Dumpcap seperti pada Gambar 3.11 nantinya akan dibuka menggunakan Wireshark untuk disaring dan diubah ke bentuk CSV. Hasil *capture* yang belum disaring akan berisi informasi dari semua paket yang telah ter-*capture* seperti yang ditunjukkan pada Gambar 3.12.



Gambar 3.12 Hasil *Capture* Yang Belum Disaring

Gambar 3.12 menunjukkan hasil *capture* yang belum disaring, yang dibuka dengan menggunakan Wireshark. Dapat dilihat pada gambar tersebut bahwa Wireshark sudah mengkategorikan data-data yang diambil menjadi 8 kolom. Pada kolom pertama adalah nomor urutan paket yang diamati. Kolom kedua berisi waktu kedatangan paket. Kolom ketiga berisi selisih antara waktu kedatangan paket saat ini dengan waktu kedatangan paket sebelumnya. Kolom keempat berisi informasi alamat IP sumber sedangkan kolom kelima berisi informasi alamat IP tujuan. Pada kolom keenam memuat

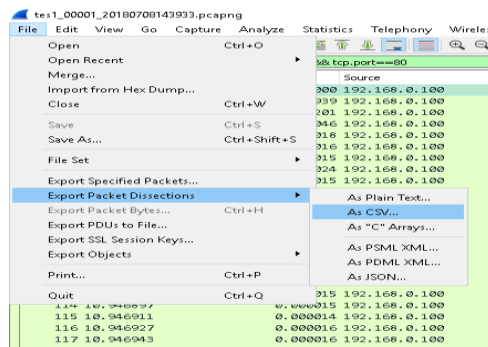
jenis protokol yang digunakan. Kolom ketujuh memuat informasi ukuran paket. Dan kolom kedelapan berisi informasi paket.



Gambar 3.13 Hasil *Capture* yang sudah disaring

Gambar 3.13 memperlihatkan hasil *capture* yang sudah disaring. Karena penelitian ini menggunakan layanan *file sharing* dan *video streaming*, maka pengiriman *file* akan menggunakan protokol TCP sedangkan video streaming menggunakan protokol UDP. Untuk itu perlu dilakukan penyaringan terhadap protokol TCP. Selain menyaring jenis protokol dan *port*-nya, dilakukan juga penyaringan terhadap alamat IP sumber dan alamat IP tujuan.

Sintaks penyaringan yang digunakan pada penelitian ini adalah “ip.src == 20.0.0.12 && ip.dst == 10.0.0.11 && tcp”. Sintaks tersebut memiliki arti untuk menyaring paket TCP yang dikirim oleh alamat IP 20.0.0.12 ke alamat IP 10.0.0.11.



Gambar 3.14 Mengekspor Hasil *Capture* Ke Bentuk CSV

Hasil *capture* yang sudah disaring kemudian diekspor ke bentuk CSV untuk diolah menjadi hasil data. Untuk mengekspornya ke bentuk CSV, cukup memilih menu “File”, kemudian memilih “Export Packet Dissection”, lalu memilih “As CSV” seperti yang terlihat pada Gambar 3.14.

3.7 Menganalisa Data

Pada tahap menganalisa data, dilakukan pengolahan data CSV yang telah diekspor menggunakan Wireshark. Pengolahan dilakukan dengan menggunakan Ms. Excel. Ada tiga hal yang dilakukan pada tahap ini, yaitu menghitung parameter *throughput*, menghitung parameter *delay*, dan menghitung parameter *packet loss*.

Perhitungan parameter *throughput* dilakukan dengan menjumlahkan ukuran paket yang diterima lalu membaginya dengan waktu pengamatan. Jumlah ukuran paket didapatkan dengan menjumlahkan seluruh nilai yang berada pada kolom *Length* dengan perintah SUM pada Ms.Excel. Sedangkan waktu pengamatan didapatkan dengan mengurangi waktu kedatangan paket terakhir dengan waktu kedatangan paket awal. Hasil data *capture wireshark* yang dirubah ke dalam format .csv terlihat seperti pada tabel 3.5.

Tabel 3.5 Hasil *capture file sharing wireshark* format .csv

A	B	C	D	E	F	G	H
No.	No.	<i>Time</i>	<i>Delta Time</i>	<i>Source</i>	<i>Destination</i>	<i>Protocol</i>	<i>Length</i>
3	6	2.40945	0	10.0.0.13	20.0.0.14	TCP	1514
4	7	2.410221	0.000771	10.0.0.13	20.0.0.14	TCP	1514
5	8	2.410222	0.000001	10.0.0.13	20.0.0.14	TCP	1514
6	9	2.410224	0.000002	10.0.0.13	20.0.0.14	TCP	1514
40	119	2.421691	0.000798	10.0.0.13	20.0.0.14	TCP	1514
41	120	2.421692	0.000001	10.0.0.13	20.0.0.14	TCP	1514
42	121	2.421693	0.000001	10.0.0.13	20.0.0.14	TCP	1514
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
21112	21119	4.633553	0.000002	10.0.0.13	20.0.0.14	TCP	1514
21113	21120	4.633555	0.000002	10.0.0.13	20.0.0.14	TCP	1514

Tabel 3.5 merupakan hasil *capture data file sharing* dengan yang digunakan untuk mendapat hasil perhitungan parameter *throughput*, *delay*, dan *packet loss*. Untuk menghitung parameter *throughput* menggunakan

penjelasan rumus 2.2 yang digunakan pada tabel 3.5 dengan jumlah data yang dikirim pada kolom H atau *Length* yaitu ukuran data dibagi dengan waktu pengiriman data pada kolom C atau *Time* dengan mengurangi waktu pengiriman akhir data dengan pengiriman awal data tersebut dikirim. Kemudian dikalikan dengan 8 untuk mengubah satuan ukuran paket dari Byte menjadi bit. Setelah itu dibagi dengan 10 pangkat 6 untuk menjadikannya ke bentuk Mbps maka didapatkan hasil parameter *throughput*.

Pada hasil data *capture wireshark* tabel 3.5 untuk menghitung hasil parameter *throughput* dengan menggunakan rumus 2.2 yang digunakan pada Ms.Excel.

$$\text{Throughput (bps)} = \frac{\text{Jumlah data yang dikirim (bit)}}{\text{waktu pengiriman data (s)}} \quad (2.2)$$

Keterangan :

- Jumlah data yang dikirim (bit) = total data yang dikirim dari *server* ke *client* (bit).
- Waktu pengiriman data = lama waktu pengiriman data dari *server* ke *client* (s).

Selanjutnya penghitungan parameter *delay* dilakukan dengan merata-ratakan *delta time* dibagi dengan jumlah total paket seperti yang ditunjukkan pada rumus 2.3 yang digunakan pada tabel 3.5.

$$\text{Delay} = \frac{\text{Total Delay}}{\text{Jumlah total paket}} \quad (2.3)$$

Keterangan :

- Total delay = Jumlah *delta time* pada sisi *client*.
- Jumlah total paket = jumlah paket yang diterima pada sisi *client*.

Penjelasan rumus 2.3 yang digunakan pada tabel 3.5 adalah kolom D dijumlahkan dibagi dengan jumlah paket paket yang diterima pada kolom B dan dikalikan dengan 1000 agar satuan *millisecond* (ms).

Lalu perhitungan parameter *packet loss* dilakukan dengan menghitung selisih dari jumlah paket yang dikirimkan dengan jumlah paket diterima kemudian dibagikan dengan jumlah paket yang dikirim. Karena parameter *packet loss* menggunakan persen (%), maka hasil yang didapat

pada perhitungan tersebut perlu dikalikan dengan 100%. Berdasarkan hasil data *capture wireshark* tabel 3.5 untuk menghitung hasil parameter *packet loss* dengan menggunakan rumus 2.5 yang digunakan pada Ms.Excel.

$$Packet Loss (\%) = \left(\frac{\text{Jumlah paket yang dikirim} - \text{Jumlah paket yang diterima}}{\text{Jumlah paket yang dikirim}} \right) \times 100\% \quad (2.5)$$

Keterangan :

- Jumlah paket yang dikirim = total paket yang dikirim *server*.
- Jumlah paket yang diterima = total paket yang diterima *client*.

Penjelasan rumus 2.5 yang digunakan pada tabel 3.5 adalah jumlah paket yang dikirim pada kolom A atau paket yang dikirim server dikurangi dengan jumlah paket yang diterima pada kolom B. Kemudian dibagi dengan jumlah paket yang dikirim pada kolom A dan dikalikan dengan 100% untuk mendapatkan persentase parameter *packet loss*.